

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application Number	09/826693	Docket Number	MIPS.0172-00-US
Filed	04/04/01	Group Art Unit	2185
Examiner	REGINALD G. BRAGDON	Customer No.	23669
Application Title	SYSTEM AND METHOD FOR DATE CACHE BYPASSING IN A STREAM PROCESSING UNIT FOR A MULTI-STREAMING PROCESSOR		
First Named Inventor	MARIO D. NEMIROVSKY		

FACSIMILE COVER SHEET

RECEIVED
CENTRAL FAX CENTER
FEB 01 2006

To: Commissioner for Patents - Central Fax Number
Fax Number: 571-273-8300

From: HUFFMAN PATENT GROUP, LLC
Fax Number: (661) 460-1986

Pages: 35 (including this cover sheet)

Dear Sir:

Please accept the attached correspondence for the above-identified matter. I hereby certify under 37 CFR 1.8 that this correspondence is being facsimile transmitted to the United States Patent and Trademark Office on the date of signature shown below.

Respectfully submitted,
HUFFMAN PATENT GROUP, LLC

/Richard K. Huffman/

By: _____

RICHARD K. HUFFMAN, P.E.
Registration No. 41,082
Tel: (719) 575-9998

02/01/2006

Date: _____

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application Number	09/826693	Docket Number	MIPS.0172-00-US
Filed	04/04/01	Group Art Unit	2185
Examiner	REGINALD G. BRAGDON	Customer No.	23669
Application Title	SYSTEM AND METHOD FOR DATE CACHE BYPASSING IN A STREAM PROCESSING UNIT FOR A MULTI-STREAMING PROCESSOR		
First Named Inventor	MARIO D. NEMIROVSKY		

LETTER TO EXAMINER

**RECEIVED
CENTRAL FAX CENTER**

FEB 01 2006

VIA FACSIMILE: 571-273-8300

Mail Stop **AMENDMENT**
Commissioner for Patents
PO Box 1450
Alexandria, VA 22313-1450

Dear Sir:

In the Office Action mailed 01/25/2006, the Examiner noted that citation AP, "Kessler et al.," from the Information Disclosure Statement submitted on 04/21/2005 was not considered because the copy of the document was not legible. The undersigned can only speculate that an illegible copy of the document was produced within the PTO. Accordingly, another copy of that portion of the submittal is attached hereto.

In addition, the undersigned notes that citation AG, "ESA/390 Principles of Operation," from the same Information Disclosure Statement, was not signed off by the Examiner. Perhaps the copy of that portion of the submittal was rendered illegible during copying as well. Accordingly, another copy of citation AG is provided herewith and follows of citation AP.

Respectfully submitted,
HUFFMAN PATENT GROUP, LLC

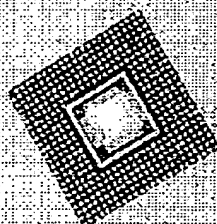
/ Richard K. Huffman /

By _____

RICHARD K. HUFFMAN
Registration No. 41,082
(719) 575-9998

02/01/2006

Date: _____



COMPAQ

The Alpha 21264 Microprocessor: Out-of-Order Execution at 600 Mhz

R. E. Kessler
COMPAQ Computer Corporation
Shrewsbury, MA

AP

PER August 1996

COMPAG

Low-latency Speculative Issue of Integer Load Data Consumers (Predict Hit)

LDQ R0, 0(R0)
ADDQ R0, R0, R1
OP R2, R2, R3

Q R E D

Q R E D

Q R E D

Q R E D

Q R E D

Q R E D

Q R E D

Q R E D

Q R E D

Q R E D

Q R E D

Q R E D

Q R E D

Q R E D

Q R E D

Q R E D

Q R E D

Q R E D

Q R E D

Q R E D

Q R E D

When predicting a load hit:

- The ADDQ issues (speculatively) after 3 cycles
- Best performance if the load actually hits (matching the prediction)
- The ADDQ issues before the load hit/miss calculation is known

If the LDQ misses when predicted to hit:

- Squash two cycles (replay the ADDQ and its consumers)
- Force a "mini-replay" (direct from the issue queue)

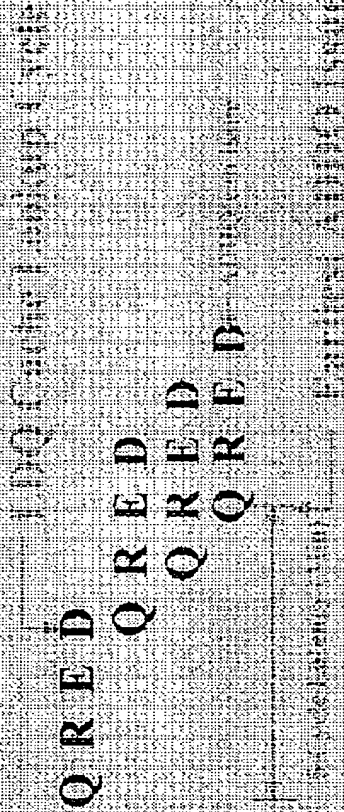
REN August 1998

19

COMPAG

Low-latency Speculative Issue of Integer Load Data Consumers (Predict Miss)

LDQ R0,0(R0) Q R E D
 OP1 R2,R2,R3 Q R E D
 OP2 R4,R4,R5 Q R E D
 ADDQ R0,R0,R1 Q R E D



When predicting a load miss:

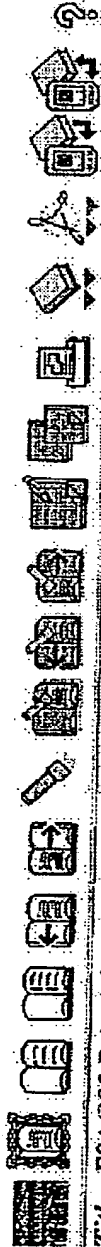
- The minimum load latency is 5 cycles (more on a miss)
- There are no squashes
- Best performance if the load actually misses (as predicted)

The hit/miss predictor:

- MSB of 4-bit counter (hits increment by 1, misses decrement by 2)

RTK August 1998

19



Title: ESA/390 Principles of Operation
 Document Number: SA22-7201-01
 Build Date: 09/23/93 08:36:13 Build Version: 1.2
 Book Path: /home/webapps/epubs/fndocs/book/dz9ar001.bo

CONTENTS Table of Contents

[Summarize]

COVER	Book Cover
NOTICES	Notices
EDITION	Edition Notice
CONTENTS	Table of Contents
FRONT_1	Notices
FRONT_1.1	Trademarks
PREFACE	Preface
PREFACE.1	Size Notation
PREFACE.2	Bytes, Characters, and Codes
PREFACE.3	Other Publications
PREFACE.4	Summary of Changes
1.0	Chapter 1. Introduction
1.1	Highlights of ESA/390
1.1.1	The ESA/370 and 370-XA Base
1.2	System Program
1.3	Compatibility
1.3.1	Compatibility among ESA/390 Systems
1.3.2	Compatibility among ESA/390, ESA/370, 370-XA, and System/370
1.3.2.1	Control-Program Compatibility
1.3.2.2	Problem-State Compatibility
1.4	Availability

AG

2/3/2006

http://publibz.boulder.ibm.com/cgi-bin/bookmgr_OS390/BOOKS/DZ9AR001/CCONTENTS

2.0	<u>Chapter 2. Organization</u>
2.1	<u>Main Storage</u>
2.2	<u>Expanded Storage</u>
2.3	<u>CPU</u>
2.3.1	<u>FSM</u>
2.3.2	<u>General Registers</u>
2.3.3	<u>Floating-Point Registers</u>
2.3.4	<u>Control Registers</u>
2.3.5	<u>Access Registers</u>
2.3.6	<u>Vector Facility</u>
2.3.7	<u>Cryptographic Facility</u>
2.4	<u>External Time Reference</u>
2.5	<u>I/O</u>
2.5.1	<u>Channel Subsystem</u>
2.5.2	<u>Channel Paths</u>
2.5.3	<u>I/O Devices and Control Units</u>
2.6	<u>Operator Facilities</u>
3.0	<u>Chapter 3. Storage</u>
3.1	<u>Storage Addressing</u>
3.1.1	<u>Information Formats</u>
3.1.2	<u>Integral Boundaries</u>
3.2	<u>Address Types and Formats</u>
3.2.1	<u>Address Types</u>
3.2.1.1	<u>Absolute Address</u>
3.2.1.2	<u>Real Address</u>
3.2.1.3	<u>Virtual Address</u>
3.2.1.4	<u>Primary Virtual Address</u>
3.2.1.5	<u>Secondary Virtual Address</u>
3.2.1.6	<u>AR-Specified Virtual Address</u>
3.2.1.7	<u>Home Virtual Address</u>
3.2.1.8	<u>Logical Address</u>
3.2.1.9	<u>Instruction Address</u>
3.2.1.10	<u>Effective Address</u>
3.2.2	<u>Address Size and Wraparound</u>
3.2.2.1	<u>Address Wraparound</u>
3.3	<u>Storage Key</u>
3.4	<u>Protection</u>
3.4.1	<u>Key-Controlled Protection</u>
3.4.1.1	<u>Storage-Protection-Override Control</u>
3.4.1.2	<u>Fetch-Protection-Override Control</u>
3.4.2	<u>Access-List-Controlled Protection</u>
3.4.3	<u>Page Protection</u>
3.4.4	<u>Low-Address Protection</u>
3.4.5	<u>Suppression on Protection</u>
3.5	<u>Reference Recording</u>

3.6	<u>Change Recording</u>
3.7	<u>Refining</u>
3.8	<u>Address Spaces</u>
	<u>Changing to Different Address Spaces</u>
	<u>Address Space Number</u>
3.9	<u>ASN Translation</u>
3.9.1	<u>ASN-Translation Controls</u>
3.9.1.1	<u>Control Register 14</u>
3.9.1.2	<u>Control Register 0</u>
3.9.2	<u>ASN-Translation Tables</u>
3.9.2.1	<u>ASN-First-Table Entries</u>
3.9.2.2	<u>ASN-Second-Table Entries</u>
3.9.3	<u>ASN-Translation Process</u>
3.9.3.1	<u>ASN-First-Table Lookup</u>
3.9.3.2	<u>ASN-Second-Table Lookup</u>
3.9.3.3	<u>Recognition of Exceptions during ASN Translation</u>
3.10	<u>ASN Authorization</u>
3.10.1	<u>ASN-Authorization Controls</u>
3.10.1.1	<u>Control Register 4</u>
3.10.1.2	<u>ASN-Second-Table Entry</u>
3.10.2	<u>Authority-Table Entries</u>
3.10.3	<u>ASN-Authorization Process</u>
3.10.3.1	<u>Authority-Table Lookup</u>
3.10.3.2	<u>Recognition of Exceptions during ASN Authorization</u>
3.11	<u>Dynamic Address Translation</u>
3.11.1	<u>Translation Control</u>
3.11.1.1	<u>Translation Modes</u>
3.11.1.2	<u>Control Register 0</u>
3.11.1.3	<u>Control Register 1</u>
3.11.1.4	<u>Control Register 7</u>
3.11.1.5	<u>Control Register 13</u>
3.11.2	<u>Translation Tables</u>
3.11.2.1	<u>Segment-Table Entries</u>
3.11.2.2	<u>Page-Table Entries</u>
3.11.2.3	<u>Summary of Segment-Table and Page-Table Sizes</u>
3.11.3	<u>Translation Process</u>
3.11.3.1	<u>Effective Segment-Table Designation</u>
3.11.3.2	<u>Inspection of Control Register 0</u>
3.11.3.3	<u>Segment-Table Lookup</u>
3.11.3.4	<u>Page-Table Lookup</u>
3.11.3.5	<u>Formation of the Real Address</u>
3.11.3.6	<u>Recognition of Exceptions during Translation</u>
3.11.4	<u>Translation-Lookaside Buffer</u>
3.11.4.1	<u>TLB Structure</u>
3.11.4.2	<u>Formation of TLB Entries</u>
3.11.4.3	<u>Use of TLB Entries</u>
3.11.4.4	<u>Modification of Translation Tables</u>

3.12	<u>Address Summary</u>
3.12.1	<u>Addresses Translated</u>
3.12.2	<u>Handling of Addresses</u>
3.13	<u>Assigned Storage Locations</u>
4.0	<u>Chapter 4: Control</u>
4.1	<u>Stopped, Operating, Load, and Check-Stop States</u>
4.1.1	<u>Stopped State</u>
4.1.2	<u>Operating State</u>
4.1.3	<u>Load State</u>
4.1.4	<u>Check-Stop State</u>
4.2	<u>Program-Status Word</u>
4.2.1	<u>Program-Status-Word Format</u>
4.3	<u>Control Registers</u>
4.4	<u>Tracing</u>
4.4.1	<u>Control Register Allocation</u>
4.4.2	<u>Trace Entries</u>
4.4.3	<u>Operation</u>
4.5	<u>Program-Event Recording</u>
4.5.1	<u>Control-Register Allocation and Segment-Table Designation</u>
4.5.2	<u>Operation</u>
4.5.2.1	<u>Identification of Cause</u>
4.5.2.2	<u>Priority of Indication</u>
4.5.3	<u>Storage-Area Designation</u>
4.5.4	<u>PER Events</u>
4.5.4.1	<u>Successful Branching</u>
4.5.4.2	<u>Instruction Fetching</u>
4.5.4.3	<u>Storage Alteration</u>
4.5.4.4	<u>General Register Alteration</u>
4.5.4.5	<u>Store Using Real Address</u>
4.5.5	<u>Indication of PER Events Concurrently with Other Interruption Conditions</u>
4.6	<u>Timing</u>
4.6.1	<u>Time-of-Day Clock</u>
4.6.1.1	<u>Format</u>
4.6.1.2	<u>States</u>
4.6.1.3	<u>Changes in Clock State</u>
4.6.1.4	<u>Setting and Inspecting the Clock</u>
4.6.2	<u>TOD-Clock Synchronization</u>
4.6.3	<u>Clock Comparator</u>
4.6.4	<u>CPU Timer</u>
4.7	<u>Externally Initiated Functions</u>
4.7.1	<u>Resets</u>
4.7.1.1	<u>CPU Reset</u>
4.7.1.2	<u>Initial CPU Reset</u>
4.7.1.3	<u>Subsystem Reset</u>
4.7.1.4	<u>Clear Reset</u>
4.7.1.5	<u>Power-On Reset</u>

4.7.2	Initial Program Loading
4.7.3	Store Status
4.8	<u>MultiProcessing</u>
4.8.1	Shared Main Storage
4.8.2	CPU Address Identification
4.9	<u>CPU Signaling and Response</u>
4.9.1	Signal-Processor Orders
4.9.2	Conditions Determining Response
4.9.2.1	Conditions Precluding Interpretation of the Order Code
4.9.2.2	Status Bits
5.0	<u>Chapter 5. Program Execution</u>
5.1	<u>Instructions</u>
5.1.1	Operands
5.1.2	<u>Instruction Formats</u>
5.1.2.1	Register Operands
5.1.2.2	Immediate Operands
5.1.2.3	Storage Operands
5.2	<u>Address Generation</u>
5.2.1	Binodal Addressing
5.2.2	Sequential Instruction-Address Generation
5.2.3	Operand-Address Generation
5.2.3.1	Formation of the Intermediate Value
5.2.3.2	Formation of the Address
5.2.4	Branch-Address Generation
5.2.4.1	Formation of the Branch Address
5.3	<u>Instruction Execution and Sequencing</u>
5.3.1	Decision Making
5.3.2	Loop Control
5.3.3	Subroutine Linkage without the Linkage Stack
5.3.4	Interruptions
5.3.5	<u>Types of Instruction Ending</u>
5.3.5.1	Completion
5.3.5.2	Suppression
5.3.5.3	Nullification
5.3.5.4	Termination
5.3.6	<u>Interruptible Instructions</u>
5.3.6.1	Point of Interruption
5.3.6.2	Unit of Operation
5.3.6.3	Execution of Interruptible Instructions
5.3.7	<u>Exceptions to Nullification and Suppression</u>
5.3.7.1	Storage Changes and Restoration for DAT-Associated Access Exceptions
5.3.7.2	Modification of DAT-Table Entries
5.3.7.3	Trial Execution for Editing Instructions and Translate Instruction
5.4	<u>Authorization Mechanisms</u>
5.4.1	Mode Requirements
5.4.2	Extraction-Authority Control

5.4.3	<u>PSM-Key Mask</u>
5.4.4	<u>Secondary-Space Control</u>
5.4.5	<u>Subsystem-Linkage Control</u>
5.4.6	<u>ASN-Translation Control</u>
5.4.7	<u>Authorization Index</u>
5.4.8	<u>Access-Register and Linkage-Stack Mechanisms</u>
5.5	<u>PC-Number Translation</u>
5.5.1	<u>PC-Number Translation Control</u>
5.5.1.1	<u>Control Register 0</u>
5.5.1.2	<u>Control Register 5</u>
5.5.2	<u>PC-Number Translation Tables</u>
5.5.2.1	<u>Linkage-Table Entries</u>
5.5.2.2	<u>Entry-Table Entries</u>
5.5.3	<u>PC-Number-Translation Process</u>
5.5.3.1	<u>Obtaining the Linkage-Table Designation</u>
5.5.3.2	<u>Linkage-Table Lookup</u>
5.5.3.3	<u>Entry-Table Lookup</u>
5.5.3.4	<u>Recognition of Exceptions during PC-Number Translation</u>
5.6	<u>Home Address Space</u>
5.7	<u>Access-Register Introduction</u>
5.7.1	<u>Summary</u>
5.7.2	<u>Access-Register Functions</u>
5.7.2.1	<u>Access-Register-Specified Address Spaces</u>
5.7.2.2	<u>Access-Register Instructions</u>
5.8	<u>Access-Register Translation</u>
5.8.1	<u>Access-Register-Translation Control</u>
5.8.1.1	<u>Address-Space-Function Control</u>
5.8.1.2	<u>Control Register 2</u>
5.8.1.3	<u>Control Register 5</u>
5.8.1.4	<u>Control Register 8</u>
5.8.2	<u>Access Registers</u>
5.8.3	<u>Access-Register-Translation Tables</u>
5.8.3.1	<u>Access-List Designations</u>
5.8.3.2	<u>Access-List Entries</u>
5.8.3.3	<u>Extended ASN-Second-Table Entries</u>
5.8.4	<u>Access-Register-Translation Process</u>
5.8.4.1	<u>Selecting the Access-List-Entry Token</u>
5.8.4.2	<u>Obtaining the Primary or Secondary Segment-Table Designation</u>
5.8.4.3	<u>Clicking the First Byte of the ALET</u>
5.8.4.4	<u>Obtaining the Effective Access-List Designation</u>
5.8.4.5	<u>Access-List Lookup</u>
5.8.4.6	<u>Locating the ASN-Second-Table Entry</u>
5.8.4.7	<u>Authorizing the Use of the Access-List Entry</u>
5.8.4.8	<u>Checking for Access-List-Controlled Protection</u>
5.8.4.9	<u>Obtaining the Segment-Table Designation from the ASN-Second-Table Entry</u>
5.8.4.10	<u>Recognition of Exceptions during Access-Register Translation</u>
5.8.5	<u>Anti-Lookaside Buffer</u>

5.8.5.1	<u>ALB Structure</u>
5.8.5.2	<u>Formation of ALB Entries</u>
5.8.5.3	<u>Modification of APT Tables</u>
5.9	<u>Linkage-Stack Introduction</u>
5.9.1	<u>Summary</u>
5.9.2	<u>Linkage-Stack Functions</u>
5.9.2.1	<u>Transferring Program Control</u>
5.9.2.2	<u>Branching Using the Linkage Stack</u>
5.9.2.3	<u>Adding and Retrieving Information</u>
5.9.2.4	<u>Testing Authorization</u>
5.9.2.5	<u>Program-Problem Analysis</u>
5.10	<u>Extended Entry-Table Entries</u>
5.11	<u>Linkage-Stack Operations</u>
5.11.1	<u>Linkage-Stack-Operations Control</u>
5.11.1.1	<u>Control Register 0</u>
5.11.1.2	<u>Control Register 15</u>
5.11.2	<u>Linkage Stack</u>
5.11.2.1	<u>Entry Descriptors</u>
5.11.2.2	<u>Header Entries</u>
5.11.2.3	<u>Trailer Entries</u>
5.11.2.4	<u>State Entries</u>
5.11.3	<u>Stacking Process</u>
5.11.3.1	<u>Locating Space for a New Entry</u>
5.11.3.2	<u>Forming the New Entry</u>
5.11.3.3	<u>Updating the Current Entry</u>
5.11.3.4	<u>Updating Control Register 15</u>
5.11.3.5	<u>Recognition of Exceptions during the Stacking Process</u>
5.11.4	<u>Unstacking Process</u>
5.11.4.1	<u>Locating the Current Entry and Processing a Header Entry</u>
5.11.4.2	<u>Checking for a State Entry</u>
5.11.4.3	<u>Restoring Information</u>
5.11.4.4	<u>Updating the Preceding Entry</u>
5.11.4.5	<u>Updating Control Register 15</u>
5.11.4.6	<u>Recognition of Exceptions during the Unstacking Process</u>
5.12	<u>Sequence of Storage References</u>
5.12.1	<u>Conceptual Sequence</u>
5.12.2	<u>Overlapped Operation of Instruction Execution</u>
5.12.3	<u>Divisible Instruction Execution</u>
5.12.4	<u>Interlocks for Virtual-Storage References</u>
5.12.4.1	<u>Interlocks between Instructions</u>
5.12.4.2	<u>Interlocks within a Single Instruction</u>
5.12.5	<u>Instruction Fetching</u>
5.12.6	<u>APT-Table and DAT-Table Fetches</u>
5.12.7	<u>Storage-Key Accesses</u>
5.12.8	<u>Storage-Operand References</u>
5.12.8.1	<u>Storage-Operand Fetch References</u>
5.12.8.2	<u>Storage-Operand Store References</u>

5.12.8.3	<u>Storage-Operand Update References</u>
5.12.9	<u>Storage-Operand Consistency</u>
5.12.9.1	<u>Single-Access References</u>
5.12.9.2	<u>Multiple-Access References</u>
5.12.9.3	<u>Block-Concurrent References</u>
5.12.9.4	<u>Consistency Specification</u>
5.12.10	<u>Relation between Operand Accesses</u>
5.12.11	<u>Other Storage References</u>
5.13	<u>Serialization</u>
5.13.1	<u>CPU Serialization</u>
5.13.2	<u>Channel-Program Serialization</u>
6.0	<u>Chapter 6. Interruptions</u>
6.1	<u>Interruption Action</u>
6.1.1	<u>Interruption Code</u>
6.1.2	<u>Enabling and Disabling</u>
6.1.3	<u>Handling of Floating Interruption Conditions</u>
6.1.4	<u>Instruction-Length Code</u>
6.1.4.1	<u>Zero ILC</u>
6.1.4.2	<u>ILC on Instruction-Patching Exceptions</u>
6.1.5	<u>Exceptions Associated with the PSW</u>
6.1.5.1	<u>Early Exception Recognition</u>
6.1.5.2	<u>Late Exception Recognition</u>
6.2	<u>External Interruption</u>
6.2.1	<u>Clock Comparator</u>
6.2.2	<u>CPU Timer</u>
6.2.3	<u>Emergency Signal</u>
6.2.4	<u>External Call</u>
6.2.5	<u>Interrupt Key</u>
6.2.6	<u>Malfunction Alert</u>
6.2.7	<u>Service Signal</u>
6.2.8	<u>TOD-Clock Sync Check</u>
6.3	<u>I/O Interruption</u>
6.4	<u>Machine-Check Interruption</u>
6.5	<u>Program Interruption</u>
6.5.1	<u>Exception-Extension Code</u>
6.5.2	<u>Program-Interruption Conditions</u>
6.5.2.1	<u>Addressing Exception</u>
6.5.2.2	<u>APX-Translation Exception</u>
6.5.2.3	<u>ALLEN-Translation Exception</u>
6.5.2.4	<u>ALB-Sequence Exception</u>
6.5.2.5	<u>ALERT-Specification Exception</u>
6.5.2.6	<u>ASN-Translation-Specification Exception</u>
6.5.2.7	<u>ASTE-Sequence Exception</u>
6.5.2.8	<u>ASTE-Validity Exception</u>
6.5.2.9	<u>ASX-Translation Exception</u>
6.5.2.10	<u>Data Exception</u>

6.5.2.11	<u>Decimal-Divide Exception</u>
6.5.2.12	<u>Decimal-Overflow Exception</u>
6.5.2.13	<u>Execute Exception</u>
6.5.2.14	<u>Exponent-Overflow Exception</u>
6.5.2.15	<u>Exponent-Underflow Exception</u>
6.5.2.16	<u>FX-Translation Exception</u>
6.5.2.17	<u>Extended-Authority Exception</u>
6.5.2.18	<u>Fixed-Point-Divide Exception</u>
6.5.2.19	<u>Fixed-Point-Overflow Exception</u>
6.5.2.20	<u>Floating-Point-Divide Exception</u>
6.5.2.21	<u>IX-Translation Exception</u>
6.5.2.22	<u>Monitor Event</u>
6.5.2.23	<u>Operand Exception</u>
6.5.2.24	<u>Operation Exception</u>
6.5.2.25	<u>Page-Translation Exception</u>
6.5.2.26	<u>PC-Translation-Specification Exception</u>
6.5.2.27	<u>PER Event</u>
6.5.2.28	<u>Primary-Authority Exception</u>
6.5.2.29	<u>Privileged-Operation Exception</u>
6.5.2.30	<u>Protection Exception</u>
6.5.2.31	<u>Secondary-Authority Exception</u>
6.5.2.32	<u>Segment-Translation Exception</u>
6.5.2.33	<u>Significance Exception</u>
6.5.2.34	<u>Space-Switch Event</u>
6.5.2.35	<u>Special-Operation Exception</u>
6.5.2.36	<u>Specification Exception</u>
6.5.2.37	<u>Square-Root Exception</u>
6.5.2.38	<u>Stack-Empty Exception</u>
6.5.2.39	<u>Stack-Full Exception</u>
6.5.2.40	<u>Stack-Operation Exception</u>
6.5.2.41	<u>Stack-Specification Exception</u>
6.5.2.42	<u>Stack-Type Exception</u>
6.5.2.43	<u>Trace-Table Exception</u>
6.5.2.44	<u>Translation-Specification Exception</u>
6.5.2.45	<u>Unnormalized-Operand Exception</u>
6.5.2.46	<u>Vector-Operation Exception</u>
6.5.3	<u>Collective Program-Interruption Names</u>
6.5.4	<u>Recognition of Access Exceptions</u>
6.5.5	<u>Multiple Program-Interruption Conditions</u>
6.5.5.1	<u>Access Exceptions</u>
6.5.5.2	<u>ASN-Translation Exceptions</u>
6.5.5.3	<u>Trace Exceptions</u>
6.6	<u>Restart Interruption</u>
6.7	<u>Supervisor-Call Interruption</u>
6.8	<u>Priority of Interruptions</u>
7.0	<u>Chapter 7. General Instructions</u>

7.1	<u>Data Format</u>
7.2	<u>Binary-Integer Representation</u>
7.3	<u>Binary Arithmetic</u>
7.3.1	<u>Signed Binary Arithmetic</u>
7.3.1.1	<u>Addition and Subtraction</u>
7.3.1.2	<u>Fixed-Point Overflow</u>
7.3.2	<u>Unsigned Binary Arithmetic</u>
7.4	<u>Signed and Logical Comparison</u>
7.5	<u>Instructions</u>
7.5.1	<u>ADD</u>
7.5.2	<u>ADD HALFWORD</u>
7.5.3	<u>ADD LOGICAL</u>
7.5.4	<u>AND</u>
7.5.5	<u>BRANCH AND LINK</u>
7.5.6	<u>BRANCH AND SAVE</u>
7.5.7	<u>BRANCH AND SAVE AND SET MODE</u>
7.5.8	<u>BRANCH AND SET MODE</u>
7.5.9	<u>BRANCH ON CONDITION</u>
7.5.10	<u>BRANCH ON COUNT</u>
7.5.11	<u>BRANCH ON INDEX HIGH</u>
7.5.12	<u>BRANCH ON INDEX LOW OR EQUAL</u>
7.5.13	<u>COMPARE</u>
7.5.14	<u>COMPARE AND FORM CODEWORD</u>
7.5.15	<u>COMPARE AND SWAP</u>
7.5.16	<u>COMPARE DOUBLE AND SWAP</u>
7.5.17	<u>COMPARE HALFWORD</u>
7.5.18	<u>COMPARE LOGICAL</u>
7.5.19	<u>COMPARE LOGICAL CHARACTERS UNDER MASK</u>
7.5.20	<u>COMPARE LOGICAL LONG</u>
7.5.21	<u>COMPARE LOGICAL STRING</u>
7.5.22	<u>COMPARE UNTIL SUBSTRING EQUAL</u>
7.5.23	<u>CONVERT TO BINARY</u>
7.5.24	<u>CONVERT TO DECIMAL</u>
7.5.25	<u>COPY ACCESS</u>
7.5.26	<u>DIVIDE</u>
7.5.27	<u>EXCLUSIVE OR</u>
7.5.28	<u>EXECUTE</u>
7.5.29	<u>EXTRACT ACCESS</u>
7.5.30	<u>INSERT CHARACTER</u>
7.5.31	<u>INSERT CHARACTERS UNDER MASK</u>
7.5.32	<u>INSERT PROGRAM MASK</u>
7.5.33	<u>LOAD</u>
7.5.34	<u>LOAD ACCESS MULTIPLE</u>
7.5.35	<u>LOAD ADDRESS</u>
7.5.36	<u>LOAD ADDRESS EXTENDED</u>
7.5.37	<u>LOAD AND TEST</u>
7.5.38	<u>LOAD COMPLEMENT</u>

7.5.39 LOAD HALFWORD
 7.5.40 LOAD MULTIPLE
 7.5.41 LOAD NEGATIVE
 7.5.42 LOAD POSITIVE
 7.5.43 MONITOR CALL
 7.5.44 MOVE
 7.5.45 MOVE INVERSE
 7.5.46 MOVE LONG
 7.5.47 MOVE NUMERICS
 7.5.48 MOVE PAGE (Facility 1)
 7.5.49 MOVE STRING
 7.5.50 MOVE WITH OFFSET
 7.5.51 MOVE ZONES
 7.5.52 MULTIPLY
 7.5.53 MULTIPLY HALFWORD
 7.5.54 OR
 7.5.55 PACK
 7.5.56 SEARCH STRING
 7.5.57 SET ACCESS
 7.5.58 SET PROGRAM MASK
 7.5.59 SHIFT LEFT DOUBLE
 7.5.60 SHIFT LEFT DOUBLE LOGICAL
 7.5.61 SHIFT LEFT SINGLE
 7.5.62 SHIFT LEFT SINGLE LOGICAL
 7.5.63 SHIFT RIGHT DOUBLE
 7.5.64 SHIFT RIGHT DOUBLE LOGICAL
 7.5.65 SHIFT RIGHT SINGLE
 7.5.66 SHIFT RIGHT SINGLE LOGICAL
 7.5.67 STORE
 7.5.68 STORE ACCESS MULTIPLE
 7.5.69 STORE CHARACTER
 7.5.70 STORE CHARACTERS UNDER MASK
 7.5.71 STORE CLOCK
 7.5.72 STORE HALFWORD
 7.5.73 STORE MULTIPLE
 7.5.74 SUBTRACT
 7.5.75 SUBTRACT HALFWORD
 7.5.76 SUBTRACT LOGICAL
 7.5.77 SUPERVISOR CALL
 7.5.78 TEST AND SET
 7.5.79 TEST UNDER MASK
 7.5.80 TRANSLATE
 7.5.81 TRANSLATE AND TEST
 7.5.82 UNPACK
 7.5.83 UPDATE TREE

8.0

Chapter 8. Decimal Instructions

8.1	<u>Decimal-Number Formats</u>
8.1.1	<u>Zoned Format</u>
8.1.2	<u>Packed Format</u>
8.1.3	<u>Decimal Codes</u>
8.2	<u>Decimal Operations</u>
8.2.1	<u>Decimal Arithmetic Instructions</u>
8.2.2	<u>Editing Instructions</u>
8.2.3	<u>Execution of Decimal Instructions</u>
8.2.4	<u>Other Instructions for Decimal Operands</u>
8.3	<u>Instructions</u>
8.3.1	<u>ADD DECIMAL</u>
8.3.2	<u>COMPARE DECIMAL</u>
8.3.3	<u>DIVIDE DECIMAL</u>
8.3.4	<u>EDIT</u>
8.3.5	<u>EDIT AND MARK</u>
8.3.6	<u>MULTIPLY DECIMAL</u>
8.3.7	<u>SHIFT AND ROUND DECIMAL</u>
8.3.8	<u>SUBTRACT DECIMAL</u>
8.3.9	<u>ZERO AND ADD</u>

9.0	<u>Chapter 9. Floating-Point Instructions</u>
9.1	<u>Floating-Point Number Representation</u>
9.2	<u>Normalization</u>
9.3	<u>Floating-Point Data Format</u>
9.4	<u>Instructions</u>
9.4.1	<u>ADD NORMALIZED</u>
9.4.2	<u>ADD UNNORMALIZED</u>
9.4.3	<u>COMPARE</u>
9.4.4	<u>DIVIDE</u>
9.4.5	<u>HALVE</u>
9.4.6	<u>LOAD</u>
9.4.7	<u>LOAD AND TEST</u>
9.4.8	<u>LOAD COMPLEMENT</u>
9.4.9	<u>LOAD NEGATIVE</u>
9.4.10	<u>LOAD POSITIVE</u>
9.4.11	<u>LOAD ROUNDED</u>
9.4.12	<u>MULTIPLY</u>
9.4.13	<u>SQUARE ROOT</u>
9.4.14	<u>STORE</u>
9.4.15	<u>SUBTRACT NORMALIZED</u>
9.4.16	<u>SUBTRACT UNNORMALIZED</u>

10.0	<u>Chapter 10. Control Instructions</u>
10.1	<u>BRANCH AND STACK</u>
10.2	<u>DIAGNOSE</u>
10.3	<u>EXTRACT PRIMARY ASN</u>
10.4	<u>EXTRACT SECONDARY ASN</u>

10.5 EXTRACT STACKED REGISTERS
 10.6 EXTRACT STACKED STATE
 10.7 INSERT ADDRESS SPACE CONTROL
 10.8 INSERT PSW KEY
 10.9 INSERT STORAGE KEY EXTENDED
 10.10 INSERT VIRTUAL STORAGE KEY
 10.11 INVALIDATE PAGE TABLE ENTRY
 10.12 LOAD ADDRESS SPACE PARAMETERS
 10.13 LOAD CONTROL
 10.14 LOAD PSW
 10.15 LOAD REAL ADDRESS
 10.16 LOAD USING REAL ADDRESS
 10.17 MODIFY STACKED STATE
 10.18 MOVE PAGE (FACILITY 2)
 10.19 MOVE TO PRIMARY
 10.20 MOVE TO SECONDARY
 10.21 MOVE WITH DESTINATION KEY
 10.22 MOVE WITH KEY
 10.23 MOVE WITH SOURCE KEY
 10.24 PROGRAM CALL
 10.25 PROGRAM RETURN
 10.26 PROGRAM TRANSFER
 10.27 PURGE ALB
 10.28 PURGE TLB
 10.29 RESET REFERENCE BIT EXTENDED
 10.30 SET ADDRESS SPACE CONTROL
 10.31 SET ADDRESS SPACE CONTROL FAST
 10.32 SET CLOCK
 10.33 SET CLOCK COMPARATOR
 10.34 SET CPU TIMER
 10.35 SET PREFIX
 10.36 SET PSW KEY FROM ADDRESS
 10.37 SET SECONDARY ASN
 10.38 SET STORAGE KEY EXTENDED
 10.39 SET SYSTEM MASK
 10.40 SIGNAL PROCESSOR
 10.41 STORE CLOCK COMPARATOR
 10.42 STORE CONTROL
 10.43 STORE CPU ADDRESS
 10.44 STORE CPU ID
 10.45 STORE CPU TIMER
 10.46 STORE PREFIX
 10.47 STORE THEN AND SYSTEM MASK
 10.48 STORE THEN OR SYSTEM MASK
 10.49 STORE USING REAL ADDRESS
 10.50 TEST ACCESS
 10.51 TEST BLOCK

10.52	<u>TEST PROTECTION</u>
10.53	<u>TRACE</u>
11.0	<u>Chapter 11. Machine-Check Handling</u>
11.1	<u>Machine-Check Detection</u>
11.2	<u>Correction of Machine Malfunctions</u>
11.2.1	<u>Error Checking and Correction</u>
11.2.2	<u>CPU Retry</u>
11.2.2.1	<u>Effects of CPU Retry</u>
11.2.2.2	<u>Checkpoint Synchronization</u>
11.2.2.3	<u>Handling of Machine Checks during Checkpoint Synchronization</u>
11.2.2.4	<u>Checkpoint-Synchronization Operations</u>
11.2.2.5	<u>Checkpoint-Synchronization Action</u>
11.2.3	<u>Channel-Subsystem Recovery</u>
11.2.4	<u>Unit Deletion</u>
11.3	<u>Handling of Machine Checks</u>
11.3.1	<u>Validation</u>
11.3.2	<u>Invalid CBC in Storage</u>
11.3.2.1	<u>Programmed Validation of Storage</u>
11.3.3	<u>Invalid CBC in Storage Keys</u>
11.3.4	<u>Invalid CBC in Registers</u>
11.4	<u>Check-Stop State</u>
11.4.1	<u>System Check Stop</u>
11.5	<u>Machine-Check Interruption</u>
11.5.1	<u>Exigent Conditions</u>
11.5.2	<u>Repressible Conditions</u>
11.5.3	<u>Interruption Action</u>
11.5.4	<u>Point of Interruption</u>
11.6	<u>Machine-Check-Interruption Code</u>
11.6.1	<u>Subclass</u>
11.6.1.1	<u>System Damage</u>
11.6.1.2	<u>Instruction-Processing Damage</u>
11.6.1.3	<u>System Recovery</u>
11.6.1.4	<u>Timing-Facility Damage</u>
11.6.1.5	<u>External Damage</u>
11.6.1.6	<u>Vector-Facility Failure</u>
11.6.1.7	<u>Degradation</u>
11.6.1.8	<u>Warning</u>
11.6.1.9	<u>Channel Report Pending</u>
11.6.1.10	<u>Service-Processor Damage</u>
11.6.1.11	<u>Channel-Subsystem Damage</u>
11.6.2	<u>Subclass Modifiers</u>
11.6.2.1	<u>Vector-Facility Source</u>
11.6.2.2	<u>Backed Up</u>
11.6.2.3	<u>Delayed Access Exception</u>
11.6.3	<u>Synchronous Machine-Check-Interruption Conditions</u>
11.6.3.1	<u>Processing Backup</u>

11.6.3.2	<u>Processing Damage</u>
11.6.4	<u>Storage Errors</u>
11.6.4.1	<u>Storage Error Uncorrected</u>
11.6.4.2	<u>Storage Error Corrected</u>
11.6.4.3	<u>Storage-Key Error Uncorrected</u>
11.6.4.4	<u>Storage Degradation</u>
11.6.4.5	<u>Indirect Storage Error</u>
11.6.5	<u>Machine-Check Interruption-Code Validity Bits</u>
11.6.5.1	<u>PSW-MWP Validity</u>
11.6.5.2	<u>PSW Mask and Key Validity</u>
11.6.5.3	<u>PSW Program-Mask and Condition-Code Validity</u>
11.6.5.4	<u>PSW-Instruction-Address Validity</u>
11.6.5.5	<u>Failing-Storage-Address Validity</u>
11.6.5.6	<u>External-Damage-Code Validity</u>
11.6.5.7	<u>Floating-Point-Register Validity</u>
11.6.5.8	<u>General-Register Validity</u>
11.6.5.9	<u>Control-Register Validity</u>
11.6.5.10	<u>Storage Logical Validity</u>
11.6.5.11	<u>Access-Register Validity</u>
11.6.5.12	<u>CPU-Timer Validity</u>
11.6.5.13	<u>Clock-Comparator Validity</u>
11.7	<u>Machine-Check Extended Interruption Information</u>
11.7.1	<u>Register-Save Areas</u>
11.7.2	<u>External-Damage Code</u>
11.7.3	<u>Failing-Storage Address</u>
11.8	<u>Handling of Machine-Check Conditions</u>
11.8.1	<u>Floating Interruption Conditions</u>
11.8.1.1	<u>Floating Machine-Check-Interruption Conditions</u>
11.8.1.2	<u>Floating I/O Interruptions</u>
11.9	<u>Machine-Check Masking</u>
11.9.1	<u>Channel-Report-Pending Subclass Mask</u>
11.9.2	<u>Recovery Subclass Mask</u>
11.9.3	<u>Degradation Subclass Mask</u>
11.9.4	<u>External-Damage Subclass Mask</u>
11.9.5	<u>Warning Subclass Mask</u>
11.10	<u>Machine-Check Logout</u>
11.11	<u>Summary of Machine-Check Masking</u>
12.0	<u>Chapter 12. Operator Facilities</u>
12.1	<u>Manual Operator</u>
12.2	<u>Basic Operator Facilities</u>
12.2.1	<u>Address-Compare Controls</u>
12.2.2	<u>Alter-and-Display Controls</u>
12.2.3	<u>Architectural-Mode Indicator</u>
12.2.4	<u>Architectural-Mode-Selection Controls</u>
12.2.5	<u>Check-Stop Indicator</u>
12.2.6	<u>IML Controls</u>

12.2.7	<u>Interrupt Key</u>
12.2.8	<u>Load Indicator</u>
12.2.9	<u>Load-Clear Key</u>
12.2.10	<u>Load-Normal Key</u>
12.2.11	<u>Load-Unit-Address Controls</u>
12.2.12	<u>Manual Indicator</u>
12.2.13	<u>Power Controls</u>
12.2.14	<u>Rate Control</u>
12.2.15	<u>Restart Key</u>
12.2.16	<u>Start Key</u>
12.2.17	<u>Stop Key</u>
12.2.18	<u>Store-Status Key</u>
12.2.19	<u>System-Reset-Clear Key</u>
12.2.20	<u>System-Reset-Normal Key</u>
12.2.21	<u>Test Indicator</u>
12.2.22	<u>TOD-Clock Control</u>
12.2.23	<u>Wait Indicator</u>
12.3	<u>Multiprocessing Configurations</u>
13.0	<u>Chapter 13. I/O Overview</u>
13.1	<u>Input/Output (I/O)</u>
13.2	<u>The Channel Subsystem</u>
13.2.1	<u>Subchannels</u>
13.3	<u>Attachment of Input/Output Devices</u>
13.3.1	<u>Channel Paths</u>
13.3.2	<u>Control Units</u>
13.3.3	<u>I/O Devices</u>
13.4	<u>I/O Addressing</u>
13.4.1	<u>Channel-Path Identifier</u>
13.4.2	<u>Subchannel Number</u>
13.4.3	<u>Device Number</u>
13.4.4	<u>Device Identifier</u>
13.5	<u>Execution of I/O Operations</u>
13.5.1	<u>Start-Function Initiation</u>
13.5.2	<u>Path Management</u>
13.5.3	<u>Channel-Program Execution</u>
13.5.4	<u>Conclusion of I/O Operations</u>
13.5.5	<u>I/O Interruptions</u>
14.0	<u>Chapter 14. I/O Instructions</u>
14.1	<u>I/O-Instruction Formats</u>
14.2	<u>I/O-Instruction Execution</u>
14.2.1	<u>Serialization</u>
14.2.2	<u>Operand Access</u>
14.2.3	<u>Condition Code</u>
14.2.4	<u>Program Exceptions</u>
14.3	<u>Instructions</u>

14.3.1	<u>CLEAR SUBCHANNEL</u>
14.3.2	<u>HALT SUBCHANNEL</u>
14.3.3	<u>MODIFY SUBCHANNEL</u>
14.3.4	<u>RESET CHANNEL PATH</u>
14.3.5	<u>RESUME SUBCHANNEL</u>
14.3.6	<u>SET ADDRESS LIMIT</u>
14.3.7	<u>SET CHANNEL MONITOR</u>
14.3.8	<u>START SUBCHANNEL</u>
14.3.9	<u>STORE CHANNEL PATH STATUS</u>
14.3.10	<u>STORE CHANNEL REPORT WORD</u>
14.3.11	<u>STORE SUBCHANNEL</u>
14.3.12	<u>TEST PENDING INTERRUPTION</u>
14.3.13	<u>TEST SUBCHANNEL</u>
15.0	<u>Chapter 15. Basic I/O Functions</u>
15.1	<u>Control of Basic I/O Functions</u>
15.1.1	<u>Subchannel-Information Block</u>
15.1.1.1	<u>Path-Management-Control Word</u>
15.1.1.2	<u>Subchannel-Status Word</u>
15.1.1.3	<u>Model-Dependent Area</u>
15.1.1.4	<u>Summary of Modifiable Fields</u>
15.2	<u>Channel-Path Allegiance</u>
15.2.1	<u>Working Allegiance</u>
15.2.2	<u>Active Allegiance</u>
15.2.3	<u>Dedicated Allegiance</u>
15.2.4	<u>Channel-Path Availability</u>
15.2.5	<u>Control-Unit Type</u>
15.3	<u>Clear Function</u>
15.3.1	<u>Clear-Function Path Management</u>
15.3.2	<u>Clear-Function Subchannel Modification</u>
15.3.3	<u>Clear-Function Signaling and Completion</u>
15.4	<u>Halt Function</u>
15.4.1	<u>Halt-Function Path Management</u>
15.4.2	<u>Halt-Function Signaling and Completion</u>
15.5	<u>Start Function and Resume Function</u>
15.5.1	<u>Start-Function and Resume-Function Path Management</u>
15.6	<u>Execution of I/O Operations</u>
15.6.1	<u>Blocking of Data</u>
15.6.2	<u>Operation-Request Block</u>
15.6.3	<u>Channel-Command Word</u>
15.6.4	<u>Command Code</u>
15.6.5	<u>Designation of Storage Area</u>
15.6.6	<u>Chaining</u>
15.6.6.1	<u>Data Chaining</u>
15.6.6.2	<u>Command Chaining</u>
15.6.7	<u>Skipping</u>
15.6.8	<u>Program-Controlled Interruption</u>

15.6.9	CCW Indirect Data Addressing
15.6.10	Suspension of Channel-Program Execution
15.6.11	Commands and Flags
15.6.12	Branching in Channel Programs
15.6.12.1	Transfer in Channel
15.6.13	Command Retry
15.7	Concluding I/O Operations during Initiation
15.8	Immediate Conclusion of I/O Operations
15.9	Concluding I/O Operations During Data Transfer
15.10	Channel-Path-Reset function
15.10.1	Channel-Path-Reset-Function Signaling
15.10.2	Channel-Path-Reset-Function-Completion Signaling
16.0	Chapter 16. I/O Interruptions
16.1	Interruption Conditions
16.1.1	Intermediate Interruption Condition
16.1.2	Primary Interruption Condition
16.1.3	Secondary Interruption Condition
16.1.4	Alert Interruption Condition
16.2	Priority of Interruptions
16.3	Interruption Action
16.4	Interruption-Response Block
16.5	Subchannel-Status Word
16.5.1	Subchannel Key
16.5.2	Suspend Control (S)
16.5.3	Extended-Status-Word Format (L)
16.5.4	Deferred Condition Code (CC)
16.5.5	Format (F)
16.5.6	Prefetch (P)
16.5.7	Initial-Status-Interruption Control (I)
16.5.8	Address-Limit-Checking Control (A)
16.5.9	Suppress-Suspended Interruption (U)
16.5.10	Subchannel-Control Field
16.5.10.1	Zero-Condition Code (Z)
16.5.10.2	Extended Control (E)
16.5.10.3	Path Not Operational (N)
16.5.10.4	Function Control (FC)
16.5.10.5	Activity Control (AC)
16.5.10.6	Status Control (SC)
16.5.11	CCW-Address Field
16.5.12	Device-Status Field
16.5.13	Subchannel-Status Field
16.5.13.1	Program-Controlled Interruption
16.5.13.2	Incorrect Length
16.5.13.3	Program Check
16.5.13.4	Protection Check
16.5.13.5	Channel-Data Check

16.5.13.6	<u>Channel-Control Check</u>
16.5.13.7	<u>Interface-Control Check</u>
16.5.13.8	<u>Chaining Check</u>
16.5.14	<u>Count Field</u>
16.6	<u>Extended-Status Word</u>
16.6.1	<u>Extended-Status Format 0</u>
16.6.1.1	<u>Subchannel Logout</u>
16.6.1.2	<u>Extended-Report Word</u>
16.6.1.3	<u>Failing-Storage Address</u>
16.6.2	<u>Extended-Status Format 1</u>
16.6.3	<u>Extended-Status Format 2</u>
16.6.4	<u>Extended-Status Format 3</u>
16.7	<u>Extended-Control Word</u>
17.0	<u>Chapter 17. I/O Support Functions</u>
17.1	<u>Channel-Subsystem Monitoring</u>
17.1.1	<u>Channel-Subsystem Timing</u>
17.1.1.1	<u>Channel-Subsystem Timer</u>
17.1.2	<u>Measurement-Block Update</u>
17.1.2.1	<u>Measurement-Block</u>
17.1.2.2	<u>Measurement-Block Origin</u>
17.1.2.3	<u>Measurement-Block Key</u>
17.1.2.4	<u>Measurement-Block Index</u>
17.1.2.5	<u>Measurement-Block-Update Mode</u>
17.1.2.6	<u>Measurement-Block-Update Enable</u>
17.1.2.7	<u>Control-Unit-Opening Measurement</u>
17.1.2.8	<u>Time-Interval Measurement Accuracy</u>
17.1.3	<u>Device-Connect-Time Measurement</u>
17.1.3.1	<u>Device-Connect-Time Measurement Mode</u>
17.1.3.2	<u>Device-Connect-Time Measurement Enable</u>
17.2	<u>Signals and Resets</u>
17.2.1	<u>Signals</u>
17.2.1.1	<u>Halt Signal</u>
17.2.1.2	<u>Clear Signal</u>
17.2.1.3	<u>Reset Signal</u>
17.2.2	<u>Resets</u>
17.2.2.1	<u>Channel-Path Reset</u>
17.2.2.2	<u>I/O-System Reset</u>
17.3	<u>Externally Initiated Functions</u>
17.3.1	<u>Initial Program Loading</u>
17.3.2	<u>Reconfiguration of the I/O System</u>
17.4	<u>Status Verification</u>
17.5	<u>Address-Limit Checking</u>
17.6	<u>Configuration Alert</u>
17.7	<u>Incorrect-Length-Indication Suppression</u>
17.8	<u>Concurrent Sense</u>
17.9	<u>Channel-Subsystem Recovery</u>

17.9.1	Channel Report
17.9.2	Channel Report Word
A.0	Appendix A. Number Representation and Instruction-Use Examples
A.1	Number Representation
A.1.1	Binary Integers
A.1.1.1	Signed Binary Integers
A.1.1.2	Unsigned Binary Integers
A.1.2	Decimal Integers
A.1.3	Floating-Point Numbers
A.1.4	Conversion Example
A.2	Instruction-Use Examples
A.2.1	Machine Format
A.2.2	Assembler-Language Format
A.2.2.1	Addressing Mode in Examples
A.3	General Instructions
A.3.1	ADD HALFWORD (AH)
A.3.2	AND (N, NC, NI, NR)
A.3.2.1	NI Example
A.3.3	Linkage Instructions (BAL, BALE, BAS, BASR, BASRM, BSM)
A.3.3.1	Other BAKR and BASR Examples
A.3.4	BRANCH AND STACK (BAKR)
A.3.4.1	BAKR Example 1
A.3.4.2	BAKR Example 2
A.3.4.3	BAKR Example 3
A.3.5	BRANCH ON CONDITION (BC, BCR)
A.3.6	BRANCH ON COUNT (BCT, BCTR)
A.3.7	BRANCH ON INDEX HIGH (BXH)
A.3.7.1	BXH Example 1
A.3.7.2	BXH Example 2
A.3.8	BRANCH ON INDEX LOW OR EQUAL (BXLE)
A.3.8.1	BXLE Example 1
A.3.8.2	BXLE Example 2
A.3.9	COMPARE AND FORM CODEWORD (CFC)
A.3.10	COMPARE HALFWORD (CH)
A.3.11	COMPARE LOGICAL (CL, CLC, CLL, CLR)
A.3.11.1	CLC Example
A.3.11.2	CLL Example
A.3.11.3	CLR Example
A.3.12	COMPARE LOGICAL CHARACTERS UNDER MASK (CLM)
A.3.13	COMPARE LOGICAL LONG (GLECL)
A.3.14	COMPARE LOGICAL STRING (CLST)
A.3.15	CONVERT TO BINARY (CVB)
A.3.16	CONVERT TO DECIMAL (CVD)
A.3.17	DIVIDE (D, DR)
A.3.18	EXCLUSIVE OR (X, XC, XI, XR)
A.3.18.1	XC Example

A.3.18.2
 A.3.19
 A.3.20
 A.3.21
 A.3.22
 A.3.23
 A.3.24
 A.3.24.1
 A.3.24.2
 A.3.25
 A.3.26
 A.3.27
 A.3.28
 A.3.29
 A.3.30
 A.3.31
 A.3.32
 A.3.33
 A.3.33.1
 A.3.34
 A.3.35
 A.3.35.1
 A.3.35.2
 A.3.36
 A.3.37
 A.3.38
 A.3.39
 A.3.40
 A.3.41
 A.3.42
 A.3.43
 A.3.44
 A.4
 A.4.1
 A.4.2
 A.4.3
 A.4.4
 A.4.5
 A.4.6
 A.4.7
 A.4.7.1
 A.4.7.2
 A.4.7.3
 A.4.7.4
 A.4.8
 A.5
 A.5.1

XI Example
 EXECUTE (EX)
 INSERT CHARACTERS UNDER MASK (ICM)
 LOAD (L, LR)
 LOAD ADDRESS (LA)
 LOAD HALFWORD (LH)
 MOVE (MVC, MVT)
 MVC Example
 MVI Example
 MOVE INVERSE (MVCIN)
 MOVE LONG (MVCL)
 MOVE NUMERICS (MVN)
 MOVE STRING (MVST)
 MOVE WITH OFFSET (MVO)
 MOVE ZONES (MVZ)
 MULTIPLY (M, MR)
 MULTIPLY HALFWORD (MH)
 OR (O, OC, OL, OR)
 OI Example
 PACK (PACK)
 SEARCH STRING (SRST)
 SRST Example 1
 SRST Example 2
 SHIFT LEFT DOUBLE (SLDA)
 SHIFT LEFT SINGLE (SLA)
 STORE CHARACTERS UNDER MASK (STCM)
 STORE MULTIPLE (STM)
 TEST UNDER MASK (TM)
 TRANSLATE (TR)
 TRANSLATE AND TEST (TRT)
 UNPACK (UNPK)
 UPDATE TREE (UPT)
 Decimal Instructions
 ADD DECIMAL (AP)
 COMPARE DECIMAL (CP)
 DIVIDE DECIMAL (DP)
 EDIT (ED)
 EDIT AND MARK (EDMK)
 MULTIPLY DECIMAL (MP)
 SHIFT AND ROUND DECIMAL (SRP)
 Decimal Left Shift
 Decimal Right Shift
 Decimal Right Shift and Round
 Multiplying by a Variable Power of 10
 ZERO AND ADD (ZAP)
 Floating-Point Instructions
 ADD NORMALIZED (AD, ADR, AE, AER, AXR)

A.5.2 ADD-UNNORMALIZED (AU, AUR, AM, AWR)
 A.5.3 COMPARE (CD, CDR, CE, CER)
 A.5.4 DIVIDE (DD, DDR, DE, DER)
 A.5.5 HALVE (HDR, HER)
 A.5.6 MULTIPLEX (MD, MDR, ME, MER, MXD, MXDR, MXR)
 A.5.7 Floating-Point-Number Conversion
 A.5.7.1 Fixed Point to Floating Point
 A.5.7.2 Floating Point to Fixed Point
 A.6 Multiprogramming and Multiprocessing Examples
 Example of a Program Failure Using OR Immediate
 Conditional Swapping Instructions (CS, CDS)
 Setting a Single Bit
 Updating Counters
 Bypassing Post and Wait
 A.6.3.1 Bypass Post Routine
 A.6.3.2 Bypass Wait Routine
 A.6.4 Lock/Unlock
 A.6.4.1 Lock/Unlock with LIFO Queuing for Contentions
 A.6.4.2 Lock/Unlock with FIFO Queuing for Contentions
 A.6.5 Free-Pool Manipulation
 A.7 Sorting Instructions
 Tree Format
 Example of Use of Sort Instructions

Appendix B. Lists of Instructions

Appendix C. Condition-Code Settings

Appendix D. Comparison between ESA/370 and ESA/390

D.0 New Facilities in ESA/390
 D.1 Access-List-Controlled Protection
 D.1.1 PER 2
 D.1.2 Concurrent Sense
 D.1.3 Storage-Protection Override
 D.1.4 Move-Page Facility 2
 D.1.5 Square Root
 D.1.6 String Instructions
 D.1.7 Suppression on Protection
 D.1.8 Set Address Space Control Fast
 D.1.9 Comparison of Facilities
 D.2

Appendix E. Comparison between 370-XA and ESA/370

E.0 New Facilities in ESA/370
 E.1 Access Registers
 E.1.1 Compare Until Substring Equal
 E.1.2 Home Address Space
 E.1.3 Linkage Stack

E.1.5
E.1.6
E.1.7
E.1.8

E.2

E.3

E.3.1

E.3.2

E.3.3

E.3.4

E.3.5

E.3.6

E.3.7

E.3.8

E.3.9

E.3.10

E.4

E.4.1

E.4.2

E.4.3

E.4.4

E.4.5

Load and Store Using Real Address
Move Page Facility 1
Move with Source or Destination Key
Private Space

Comparison of Facilities

Summary of Changes

New Instructions Provided

Comparison of PSW Formats

New Control-Register Assignments

New Assigned Storage Locations

New Exceptions

Change to Secondary-Space Mode

Changes to ASN-Second-Table Entry and ASN Translation

Changes to Entry-Table Entry and PC-Number Translation

Changes to Program Call

Changes to Set Address Space Control

Effects in New Translation Modes

Effects on Interlocks for Virtual-Storage References

Effect on Inhibit Address Space Control

Effect on Load Real Address

Effect on Test Pending Interruption

Effect on Test Protection

F.0

F.1

F.1.1

F.1.2

F.1.3

F.1.4

F.1.5

F.1.6

F.1.7

F.2

F.3

F.3.1

F.3.2

F.3.3

F.3.4

F.3.5

F.3.6

F.3.7

F.3.8

F.3.9

F.3.10

G.0

Appendix F. Comparison between System/370 and 370-XA

New Facilities in 370-XA

Unimodal Addressing

31-Bit Logical Addressing

31-Bit Real and Absolute Addressing

Page Protection

Tracing

Incorrect-Length-Indication Suppression

Status Verification

Comparison of Facilities

Summary of Changes

Changes in Instructions Provided

Input/Output Comparison

Comparison of PSW Formats

Changes in Control-Register Assignments

Changes in Assigned Storage Locations

Signal Processor Changes

Machine-Check Changes

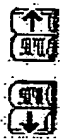
Changes to Addressing Wraparound

Changes to Load Real Address

Changes to 31-Bit Real Operand Addresses

Appendix G. Table of Powers of 2

H.O. [Appendix H. Hexadecimal Tables](#)
I.O. [Appendix I. EBCDIC Chart](#)
INDEX [Index](#)



© Copyright IBM Corp. 1990, 1991, 1993

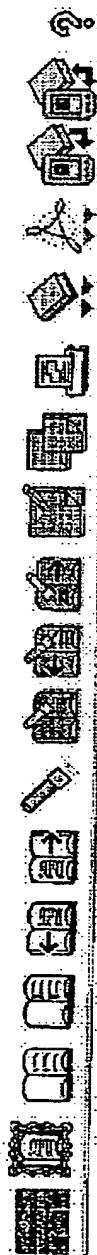
IBM Library Server® Copyright 1989, 2000 IBM Corporation. All rights reserved.

[Return to library](#): [[z/OS](#) | [z/OS.e](#) | [OS/390](#) | [TPF](#) | [VM](#) | [VSE](#) | [IBM Hardware](#) | [S/390 Redbooks](#)]
[Glossary](#): [[IBM terminology](#)]
[Publications](#): [[How to order publications](#) | [Subscribe to publication updates](#)]
[Readers](#): [[Download IBM Library Reader](#) | [Download Adobe® Acrobat® Reader®](#)]
[Contacts](#): [[Contact z/OS](#) | [Contact OS/390](#)]

Adobe, the Adobe logo, Acrobat, the Acrobat logo, and Acrobat Reader are registered trademarks of Adobe Systems Incorporated.

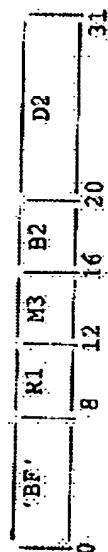
http://publibz.boulder.ibm.com/cgi-bin/bookmgr_OS390/BOOKS/DZ9AR001/CONTENTS

2/1/2006



7.5.31 INSERT CHARACTERS UNDER MASK

TCM R1, M3, D2 (B2) [RS]



Bytes from contiguous locations beginning at the second-operand address are inserted into general register R1 under control of a mask.

The contents of the M3 field are used as a mask. These four bits, left to right, correspond one for one with the four bytes, left to right, of general register R1. The byte positions corresponding to ones in the mask are filled, left to right, with bytes from successive storage locations beginning at the second-operand address. When the mask is not zero, the length of the second operand is equal to the number of ones in the mask. The bytes in the general register corresponding to zeros in the mask remain unchanged.

The resulting condition code is based on the mask and on the value of the bits inserted. When the mask is zero or when all inserted bits are zeros, the condition code is set to 0. When the inserted bits are not all zeros, the code is set according to the leftmost bit of the storage operand: if this bit is one, the code is set to 1; if this bit is zero, the code is set to 2.

When the mask is not zero, exceptions associated with storage operand access are recognized only for the number of bytes specified by the mask. When the mask is zero, access exceptions are recognized for one byte at the second-operand address.

Resulting Condition Code:

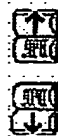
- 0 All inserted bits zeros, or mask bits all zeros
- 1 Leftmost inserted bit one
- 2 Leftmost inserted bit zero, and not all inserted bits zeros
- 3

Program Exceptions:

- 0 Access (fetch, operand 2)

Programming Notes:

1. Examples of the use of the INSERT CHARACTERS UNDER MASK instruction are given in Appendix A, "Number Representation and Instruction-Use Examples."
2. The condition code for INSERT CHARACTERS UNDER MASK is defined such that, when the mask is 1111, the instruction causes the same condition code to be set as for LOAD AND TEST. Thus, the instruction may be used as a storage-to-register load-and-test operation.
3. INSERT CHARACTERS UNDER MASK with a mask of 1111 or 0001 performs a function similar to that of a LOAD (L) or INSERT CHARACTER (IC) instruction, respectively, with the exception of the condition code setting. However, the performance of INSERT CHARACTERS UNDER MASK may be slower.



© Copyright IBM Corp. 1990, 1991, 1993

IBM Library Server® Copyright 1989, 2003 IBM Corporation. All rights reserved.

Return to library: [[z/OS](#) | [z/OS.e](#) | [OS/390](#) | [TPF](#) | [VM](#) | [VSE](#) | [IBM Hardware](#) | [S/390 Redbooks](#)]

Glossary: [[IBM terminology](#)]

Publications: [[How to order publications](#) | [Subscribe to publication updates](#)]

Readers: [[Download IBM Library Reader](#) | [Download Adobe® Acrobat® Reader®](#)]

Contacts: [[Contact z/OS](#) | [Contact OS/390](#)]

Adobe, the Adobe logo, Acrobat, the Acrobat logo, and Acrobat Reader are registered trademarks of Adobe Systems Incorporated.

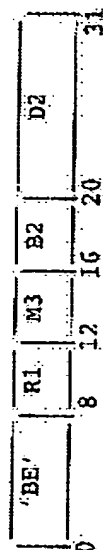
http://publibz.boulder.ibm.com/cgi-bin/bookmgr_OS390/BOOKS/DZ9AR001/7.5.31?DT=19930923083613

2/1/2006



7.5.70 STORE CHARACTERS UNDER MASK

STCM R1,M3,D2(B2) [RS]



Bytes selected from general register R1 under control of a mask are placed at contiguous byte locations beginning at the second-operand address.

The contents of the M3 field are used as a mask. These four bits, left to right, correspond one for one with the four bytes, left to right, of general register R1. The bytes corresponding to ones in the mask are placed in the same order at successive and contiguous storage locations beginning at the second-operand address. When the mask is not zero, the length of the second operand is equal to the number of ones in the mask. The contents of the general register remain unchanged.

When the mask is not zero, exceptions associated with storage-operand accesses are recognized only for the number of bytes specified by the mask.

When the mask is zero, the single byte designated by the second-operand address remains unchanged; however, on some models, the value may be fetched and subsequently stored back unchanged at the same storage location. This update appears to be an interleaved-update reference as observed by other CPUs.

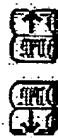
Condition Code: The code remains unchanged.

Program Exceptions:

- o Access (store, operand 2)

Programming Notes:

1. An example of the use of the STORE CHARACTERS UNDER MASK instruction is given in Appendix A, "Number Representation and Instruction-Use Examples."
2. STORE CHARACTERS UNDER MASK with a mask of 0111 may be used to store a three-byte address, for example, in modifying the address in a QCM.
3. STORE CHARACTERS UNDER MASK with a mask of 1111, 0011, or 0001 performs the same function as STORE, STORE HALFWORD, or STORE CHARACTER, respectively. However, on most models, the performance of STORE CHARACTERS UNDER MASK is slower.
4. Using STORE CHARACTERS UNDER MASK with a zero mask should be avoided since this instruction, depending on the model, may perform a fetch and store of the single byte designated by the second-operand address. This reference is not interlocked against accesses by channel programs. In addition, it may cause any of the following to occur for the byte designated by the second-operand address: a PER storage-alteration event may be recognized; access exceptions may be recognized; and, provided no access exceptions exist, the change bit may be set to one. Because the contents of storage remain unchanged, the change bit may or may not be one when a PER storage-alteration event is recognized.



© Copyright IBM Corp. 1990, 1991, 1993

IBM Library Service © Copyright 1989, 2003 IBM Corporation. All rights reserved.

Return to library: [[z/OS](#) | [z/OS.e](#) | [OS/390](#) | [TPF](#) | [VM](#) | [VSE](#) | [IBM Hardware](#) | [S/390 Redbooks](#)]

Glossary: [[IBM terminology](#)]

Publications: [[How to order publications](#) | [Subscribe to publication updates](#)]

Readers: [[Download IBM Library Reader](#) | [Download Adobe® Acrobat® Reader®](#)]

Contacts: [[Contact z/OS](#) | [Contact OS/390](#)]

Adobe, the Adobe logo, Acrobat, the Acrobat logo, and Acrobat Reader are registered trademarks of Adobe Systems Incorporated.

1.5.10 ES/VS390 Principles of Operation IBM Library Server

http://publibz.boulder.ibm.com/cgi-bin/bookmgr_OS390/BOOKS/DZ9AR001/7.5.70?DT=19930923083613

2/3/2004